

최종 산출물

GitLab 소스 클론 이후 빌드 및 배포 가이드


✓ 1. 프로젝트 개요

- 서비스명: SeeThrough
- 기술스택: Spring Boot, React, FastAPI, Tensorflow, Docker, Jenkins, PostgreSQL, Nginx
- 구성요소:
 - Frontend: React + TypeScript
 - Backend: Spring Boot + Gradle
 - Android App: Android Studio
 - Infra: Docker, Jenkins, Nginx, PostgreSQL
 - Vision 서버: 얼굴 인식 (FastAPI + Uvicorn)
 - LLM 서버: 텍스트 생성 (FastAPI + Uvicorn)

2. 환경 정보 및 버전

| 항목 | 버전 |
|----------------|--------------------------------------|
| JVM | Java 17 (openjdk:17-alpine) |
| Spring Boot | 3.1.0 |
| React | 18+ |
| Nginx | 1.24.0 (nginx:alpine 기준) |
| Gradle | 8.4 (<code>./gradlew build</code>) |
| Docker | 24.0.2 |
| Docker Compose | v2.18.1 |
| Jenkins | 2.426.1 |
| Python | 3.10 (Vision, LLM 서버) |

3. 빌드 시 사용되는 환경 변수

 `.env` 또는 `docker-compose.yml` 내 주요 환경 변수 예시

```
env
복사편집
# DB 설정
DB_URL=jdbc:postgresql://db:5432/see_through
DB_USERNAME=${DB_USERNAME}
DB_PASSWORD=${DB_PASSWORD}
DB_NAME=see_through

# 외부 API
LLM_URL=http://llm:8000
NICKNAME_URL=https://nickname.hwanmoo.kr

# Swagger 인증
SWAGGER_USERNAME=${SWAGGER_USERNAME}
SWAGGER_PASSWORD=${SWAGGER_PASSWORD}

# 알림
ALARM_UUID=00000000-0000-0000-0000-000000000001

# Vision 서버
VISION_SERVER_URL=http://vision:9000

# 프론트엔드 API 주소
VITE_API_SERVER_URL=https://j12s002.p.ssafy.io/api
VITE_LOCAL_SERVER_URL=http://localhost:9000
VITE_WS_LOCAL_SERVER_URL=ws://localhost:9000/vision/find-faces

# OpenAI & Typecast
VITE_OPENAI_API_KEY=sk-proj-...
OPENAI_API_KEY=sk-proj-...
TYPECAST_API_KEY=__pltNfk...
TYPECAST_BASE_URL=https://typecast.ai

# Firebase
```

```
FCM_TOKEN_PATH=src/main/resources/firebase/seethrough-*.json
```

! 4. 배포 시 특이사항

- 프론트엔드와 백엔드는 Docker 기반으로 실행 (`docker-compose up -d --build`)
- 백엔드 빌드: `/api` 디렉토리에서 `./gradlew build` 실행
- Vision 서버는 Python 기반 별도 서버, 로컬 또는 컨테이너로 실행 가능
- LLM 서버도 Python 기반 FastAPI 서버로 별도 실행 가능
- Swagger 인증 필요 (ID: `s002seethrough` , PW: `seethru002!`)
- Firebase 푸시 기능 사용 시 인증 JSON 필요

📁 5. 주요 설정 파일 목록

| 파일 경로 | 설명 |
|-------------------------------------------------|---------------------------------|
| <code>src/main/resources/application.yml</code> | Spring Boot 설정 |
| <code>docker-compose.yml</code> | 전체 컨테이너 정의 |
| <code>.env</code> , <code>.env.dev</code> | 환경 변수 정의 |
| <code>nginx.conf</code> | 프록시 및 SSL 설정 |
| <code>Dockerfile</code> | 각 서비스 이미지 빌드 명세 |
| <code>src/main/resources/firebase/*.json</code> | Firebase 인증 키 |
| <code>vision/run_vision.sh</code> | Vision 서버 실행 스크립트 (Linux/macOS) |
| <code>vision/requirements-mac.txt</code> | Vision 서버 의존성 목록 |
| <code>vision/face_api.py</code> | Vision 서버 진입점 |
| <code>llm/run_llm.bat</code> | LLM 서버 실행 스크립트 (Windows) |
| <code>llm/requirements.txt</code> | LLM 서버 의존성 목록 |
| <code>llm/app/main.py</code> | LLM 서버 진입점 |

👁 Vision 서버 실행 방법 (얼굴 인식)

위치: `vision/`

환경: Python 3.10 이상, Uvicorn

```

bash
vision
# 실행 스크립트 (run_vision.sh)
#!/bin/bash

echo "New Vision 서버 실행 준비중..."

# 가상환경 생성
if [ ! -d ".venv" ]; then
    echo "가상 환경을 생성합니다."
    python3.10 -m venv .venv
fi

# 가상환경 활성화
source .venv/bin/activate

# 의존성 설치
pip install -r requirements-mac.txt

# Vision 서버 실행
python -m uvicorn face_api:app --host 0.0.0.0 --port 9000

# 가상환경 비활성화
deactivate

```

실행 명령어:

```

bash
vision
cd vision
bash run_vision.sh

```

LLM 서버 실행 방법 (텍스트 생성)

위치: 

환경: Python 3.10 이상, FastAPI

```
bat
llm
:: 실행 스크립트 (run_llm.bat)
@echo off
chcp 65001 > nul
echo "FastAPI 서버 실행 준비중..."

:: 가상환경 생성
if not exist venv\ (
    echo "가상 환경을 생성합니다..."
    python -m venv venv
    echo "가상 환경 생성 완료!"
)

:: 가상환경 활성화
call venv\Scripts\activate.bat

:: 패키지 설치
if exist requirements.txt (
    echo "필요한 패키지를 설치합니다..."
    pip install -r requirements.txt
    echo "패키지 설치 완료!"
)

:: FastAPI 서버 실행
echo "FastAPI 서버를 실행합니다..."
python -m uvicorn app.main:app --reload

:: 서버 종료 후 가상 환경 비활성화
call venv\Scripts\deactivate.bat
echo "FastAPI 서버가 종료되었습니다."
pause
```

실행 명령어 (Windows):

```
cmd
llm
cd llm
run_llm.bat
```

외부 서비스 설정 요약

| 서비스명 | 설명 | 인증/접근 정보 |
|----------|-----------|---------------------------------------------------------------------------|
| OpenAI | 텍스트 생성 | API Key: <code>sk-proj-...</code> |
| Typecast | 음성 합성 TTS | API Key: <code>__pltNFk...</code> , URL: <code>https://typecast.ai</code> |
| Firebase | 푸시 알림 | 인증 키: <code>firebase/seethrough-*.json</code> |
| 닉네임 API | 랜덤 닉네임 생성 | <code>https://nickname.hwanmoo.kr</code> |